Block Tensor Decomposition for Source Apportionment of Air Pollution*

Philip K. Hopke† Maggie Leung† Na Li‡ Carmeliza Navasca‡§

Abstract

The ambient particulate chemical composition data with three particle diameter sizes (2.5mm $< D_p <$ 1.15 mm, 1.15mm $< D_p <$ 0.34mm and 0.34mm $< D_p <$ 0.1mm) collected at a major industrial center in Allen Park in Detroit, MI is examined. Standard multiway (tensor) methods like PARAFAC and Tucker tensor decompositions have been applied extensively to many chemical data. However, for multiple particle sizes, the source apportionment analysis calls for a novel multiway factor analysis. We apply the regularized block tensor decomposition to the collected air sample data. In particular, we use the Block Term Decomposition (BTD) in rank-$(L; L; 1)$ form to identify nine pollution sources (Fe+Zn, Sulfur with Dust, Road Dust, two types of Metal Works, Road Salt, Local Sulfate, and Homogeneous and Cloud Sulfate).

1 Introduction

Source apportionment analysis determines the sources of various air pollutants at a particular location. Through a collected sample of environmental data, the ambient particulate chemical composition data is acquired and then analyzed. This technique of identifying and quantifying the sources of air pollutants at a receptor location by resolving the mixture of chemical species into the contributions from the individual source types is called receptor modeling when factor analysis used. Factor analysis utilizes matrix methods, like PCA and PMF. In the study of source apportionment of airborne particles, the measured chemical composition data from the collected samples form a matrix in terms of chemical species and time samples which can be decomposed into two matrices representing sources contributions and sources profiles. Matrix methods have been a very effective tool for source apportionment for collected samples of fine particles as well as coarse particles; see e.g. [22, 13, 15]. However, chemical data can have more attributes, such as particle size distribution which is common in identification problems of air pollution sources [24]. Thus, a two-way analysis is not sufficient. Multiway factor analysis (tensor decompositions) can provide better estimations.

We examine the ambient particulate data [24] collected at Allen Park in Detroit, MI between February and April 2002. It follows the following three-way receptor model:

$$x_{ijk} = \sum_{p=1}^{P} a_{ip} b_{jpk} + e_{ijk}$$

where the measured data x_{ijk} is the concentration value of the ith sample of the kth particle size range $(0.1 - 0.34\mu m, 0.34 - 1.15\mu m, 1.15 - 2.5\mu m)$ of the jth species, a_{ip} is the pth source mass contribution during the time units for the ith sample, b_{ijk} is the jth species mass fraction of the kth particle size range from the pth source, e_{ijk} is the measurement error, and P is the total number of independent sources.

In this paper, we reformulate the three-way receptor model in a block tensor structure format and utilize a new regularized tensor algorithm found in [20]. Previous tensor applications in chemometrics or environmental data analysis have been limited to tensor CANDECOMP/PARAFAC (CP) decomposition [4, 10, 2, 3, 11, 12] or tensor Tucker decomposition [25, 26, 8]. Instead we focus on a more general type of tensor decomposition - Block Term Decomposition (BTD) [5, 6, 7], and its application in environmental pollution. In fact, the data model fits the BTD in rank-$(L, L, 1)$.

Here we apply the regularized alternating method (RALS) [20] for the BTD problems. Then, the method is used to analyze a real-life air sample tensor dataset. We show the efficacy of the regularized alternating algorithm (BTD-RALS) for solving BTD through numerical examples as well as prove a convergence property of the algorithm. In the application part, we show that the air sample data model in [24] follows the BTD format, so that the BTD-RALS method is used to identify all the different factor pollution sources.

The paper is organized as follows. We introduce the notation in Section 2. In Sections 3 and 4, we describe some background of the block term decomposition and

*Supported by the NSF DMS-0915100 and the Clarkson's Institute for a Sustainable Environment.

†Department of Chemical Engineering, Clarkson University, Potsdam, NY 13699.

‡Department of Mathematics, Clarkson University, Potsdam, NY 13699.

§Corresponding author. Email: cnavasca@clarkson.edu.

then discuss a convergence property of the algorithm. The real data example is given in Section 6. Concluding remarks are discussed in Section 6.

2 Preliminaries

We denote the scalars in $\mathbb{R}$ with lower-case letters $(a, b, \ldots)$ and the vectors with bold lower-case letters $(\mathbf{a}, \mathbf{b}, \ldots)$. The matrices are written as bold upper-case letters $(\mathbf{A}, \mathbf{B}, \ldots)$ and the symbol for tensors are calligraphic letters $(\mathcal{A}, \mathcal{B}, \ldots)$. The subscripts represent the following scalars: $(\mathcal{A})_{ijk} = a_{ijk}$, $(\mathbf{A})_{ij} = a_{ij}$, $(\mathbf{a})_i = a_i$ and $\mathbf{A}_r$ is the rth column of $\mathbf{A}$. The superscripts indicate the length of the vector or the size of the matrices. For example, $\mathbf{b}^K$ is a vector with length K and $\mathbf{B}^{N\times K}$ is a $N \times K$ matrix. In addition, the lower-case superscripts on a matrix indicate the mode in which it has been matricized. For example, $\mathbf{T}_{(n)}$ is the mode-n matricization of the tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$ for $n = 1, 2, 3$.

DEFINITION 2.1. *The Kronecker product of matrices* $\mathbf{A}$ *and* $\mathbf{B}$ *is defined as*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \ldots \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}.$$

DEFINITION 2.2. *The Khatri-Rao product is the "matching columnwise" Kronecker product. Given matrices* $\mathbf{A} \in \mathbb{R}^{I\times K}$ *and* $\mathbf{B} \in \mathbb{R}^{J\times K}$*, their Khatri-Rao product is denoted by* $\mathbf{A} \odot \mathbf{B}$*. The result is a matrix of size* $(IJ \times K)$ *defined by*

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{A}_1 \otimes \mathbf{B}_1 \quad \ldots \quad \mathbf{A}_K \otimes \mathbf{B}_K].$$

If $\mathbf{a}$ *and* $\mathbf{b}$ *are vectors, then the Khatri-Rao and Kronecker products are identical, i.e.,* $\mathbf{a} \otimes \mathbf{b} = \mathbf{a} \odot \mathbf{b}$.

DEFINITION 2.3. *Let* $\mathbf{A} = [\mathbf{A}_1 \ \ldots \ \mathbf{A}_R]$ *and* $\mathbf{B} = [\mathbf{B}_1 \ \ldots \ \mathbf{B}_R]$ *be two partitioned matrices. Then we define a product of* $\mathbf{A}$ *and* $\mathbf{B}$*, denoted* $\odot_p$*, which is*

$$\mathbf{A} \odot_p \mathbf{B} = [\mathbf{A}_1 \otimes \mathbf{B}_1 \ \ldots \ \mathbf{A}_R \otimes \mathbf{B}_R].$$

DEFINITION 2.4. (MODE-n MATRICIZATION) *Matricization is the process of reordering the elements of an* N*th order tensor into a matrix. The mode-n matricization of a tensor* $\mathcal{T} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ *is denoted by* $\mathbf{T}_{(\mathbf{n})}$ *and arranges the mode-n fibers, the vectors obtained from fixing every index with the exception of the* n*th mode, as the columns of the resulting matrix.*

DEFINITION 2.5. (VECTORIZATION) *The vectorization of a matrix* $\mathbf{M} = [\mathbf{m}_1 \ \cdots \ \mathbf{m}_n] \in \mathbb{R}^{m\times n}$*, where* $\mathbf{m}_i$ *is the* i*th column of* $\mathbf{M}$*, is denoted by* $vec(\mathbf{M})$ *which is a vector of size* mn *defined by*

$$vec(\mathbf{M}) = \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_n \end{bmatrix}.$$

DEFINITION 2.6. (FROBENIUS-NORM) *The Frobenius norm of a tensor* $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ *is the square root of the sum of the squares of all its elements. The formula is*

$$\|\mathcal{X}\|_F = \sqrt{\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\cdots\sum_{i_N=1}^{I_N} x_{i_1 i_2\cdots i_N}^2}.$$

3 Block Term Decompositions

Let $\mathcal{X}$ be a real-valued third-order tensor of size $I \times J \times K$. A decomposition of $\mathcal{X}$ in a sum of rank-$(L_r, L_r, 1)$ terms with $1 \le r \le R$ is a decomposition of the form

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{E}_r \circ \mathbf{c}_r, \tag{3.1}$$

in which the rank of the matrices $\mathbf{E}_r \in \mathbb{R}^{I\times J}$ is L_r and $\mathbf{c}_r$ are vectors of length K. Elementwise, the decomposition is defined as

$$x_{ijk} = \sum_{r=1}^{R} e_{ij}^{(r)} \cdot c_k^{(r)}.$$

So $\mathcal{X}$ is decomposed into a sum of matrix-vector outer products. If we decomposes $\mathbf{E}_r$ into two matrices, i.e., $\mathbf{E}_r = \mathbf{A}_r \cdot \mathbf{B}_r^T$, where the matrix $\mathbf{A}_r \in \mathbb{R}^{I\times L_r}$ and the matrix $\mathbf{B}_r \in \mathbb{R}^{J\times L_r}$ are rank L_r, then the equation (3.1) can be written as the following formula,

$$\mathcal{X} = \sum_{r=1}^{R} (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r. \tag{3.2}$$

We define $\mathbf{A} = [\mathbf{A}_1 \ \cdots \ \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1 \ \cdots \ \mathbf{B}]$, $\mathbf{C} = [\mathbf{c}_1 \ \cdots \ \mathbf{c}_R]$ and call them the factor matrices of the BTD-$(L_r, L_r, 1)$, where $\mathbf{A}_r$ and $\mathbf{B}_r$ are submatrices with $r = 1, \ldots, R$. When $L_r \equiv L$ for $1 \le r \le R$, the decomposition is called BTD in rank-$(L, L, 1)$. We will focus on this type of factorization and analyze the environmental dataset with BTD-$(L, L, 1)$.

See Figure 1 for the BTD in rank-$(L, L, 1)$ for a third-order tensor.

So, for a given tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$, the BTD-$(L, L, 1)$ will minimize the error function

$$f(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \left\| \mathcal{X} - \sum_{r=1}^{R} (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r \right\|_F^2. \tag{3.3}$$

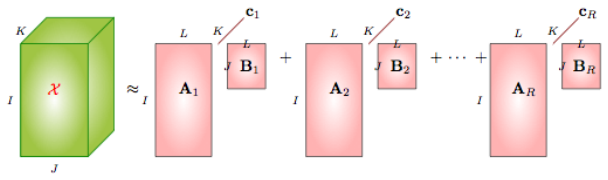


Figure 1: **BTD-**$(L, L, 1)$ **for** $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$

The general case of the block term decomposition is called BTD in rank-(L, M, N). It decomposes a tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$ in a sum of rank-(L, M, N) terms of the form

$$\mathcal{X} = \sum_{r=1}^{R} \mathcal{D}_r \times_1 \mathbf{A}_r \times_2 \mathbf{B}_r \times_3 \mathbf{C}_r, \tag{3.4}$$

in which $\mathcal{D}_r \in \mathbb{R}^{L\times M\times N}$ and where $\mathbf{A}_r \in \mathbb{R}^{I\times L}$, $\mathbf{B}_r \in \mathbb{R}^{J\times M}$ and $\mathbf{C}_r \in \mathbb{R}^{K\times N}$ are full column rank, $1 \le r \le R$.

The product '$\times_1$', '$\times_2$' and '$\times_3$' are called Tucker product which defines the multiplication between a tensor and a matrix. Figure 2 shows the BTD in rank-(L, M, N) for a third-order tensor.

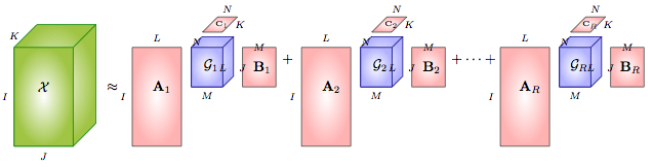


Figure 2: **BTD-**(L, M, N) **for** $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$

Actually, each term of the equation (3.4) is a Tucker model [25]. We see here if let $N = 1$, then each core tensor becomes a matrix, thereby reducing the general model to BTD-$(L_r, L_r, 1)$. If the ranks of the matrices for each term are same, then it becomes the BTD-$(L, L, 1)$. Furthermore, if $L = M = N = 1$, then the each core tensor is just a scalar and all the matrices $\mathbf{A}_r$, $\mathbf{B}_r$ and $\mathbf{C}_r$ are vectors. So, the BTD-$(1, 1, 1)$ is exactly a CP decomposition,

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r. \tag{3.5}$$

The BTD in rank-$(L, L, 1)$ is a general case of the CP decomposition. Thus, the standard and regularized algorithms for solving for CP can be applied to the BTD case. More details on the algorithms are discussed in the next section.

4 Algorithm for Block Term Decomposition

For the BTD-$(L, L, 1)$, equation (3.2) can be expressed to three equivalent equations. If we take the three different modes matricization on equation (3.2), then we have

$$\begin{aligned}
\mathbf{X}_{(1)} &= \mathbf{A}(\mathbf{C} \odot_p \mathbf{B})^T, \\
\mathbf{X}_{(2)} &= \mathbf{B}(\mathbf{C} \odot_p \mathbf{A})^T, \\
\mathbf{X}_{(3)} &= \mathbf{C}[(\mathbf{B}_1 \odot \mathbf{A}_1)\mathbf{1}_L \ \cdots \ (\mathbf{B}_R \odot \mathbf{A}_R)\mathbf{1}_L]^T,
\end{aligned}$$

where $\mathbf{1}_L$ is a column vector of all ones of length L.

The paper of De Lathauwer and Nion [7] proposes an algorithm to solve the block term decomposition in rank $(L, L, 1)$ for a third-order tensor, called the alternating least-squares for BTD (BTD-ALS).

Given a third-order tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$, our problem is to minimize the function (3.3), i.e.,

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \left\| \mathcal{X} - \sum_{r=1}^{R} (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r \right\|_F^2, \tag{4.6}$$

with respect to the factor matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$.

From the three equations above, we obtain the following three expressions for (4.6):

$$\begin{aligned}
&\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \|\mathbf{X}_{(1)} - \mathbf{A}(\mathbf{C} \odot_p \mathbf{B})^T\|_F^2, \\
&\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \|\mathbf{X}_{(2)} - \mathbf{B}(\mathbf{C} \odot_p \mathbf{A})^T\|_F^2, \\
&\text{and} \\
&\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \|\mathbf{X}_{(3)} - \mathbf{C}[(\mathbf{B}_1 \odot \mathbf{A}_1)\mathbf{1}_L \cdots (\mathbf{B}_R \odot \mathbf{A}_R)\mathbf{1}_L]^T\|_F^2.
\end{aligned}$$

These three are equivalent. Instead of solving (4.6) for the three factors one time, we can use these three equations by fixing all factor matrices but one factor each time. Then the problem reduces to three coupled linear least-squares subproblems. We have

$$\begin{aligned}
\mathbf{A}^{k+1} &= \underset{\widehat{\mathbf{A}}\in\mathbb{R}^{I\times LR}}{\operatorname{argmin}} \|\mathbf{X}_{(1)} - \widehat{\mathbf{A}}(\mathbf{C}^k \odot_p \mathbf{B}^k)^T\|_F^2, \\
\mathbf{B}^{k+1} &= \underset{\widehat{\mathbf{B}}\in\mathbb{R}^{J\times LR}}{\operatorname{argmin}} \|\mathbf{X}_{(2)} - \widehat{\mathbf{B}}(\mathbf{C}^k \odot_p \mathbf{A}^{k+1})^T\|_F^2, \\
\mathbf{C}^{k+1} &= \underset{\widehat{\mathbf{C}}\in\mathbb{R}^{K\times R}}{\operatorname{argmin}} \|\mathbf{X}_{(3)} - \widehat{\mathbf{C}}[(\mathbf{B}_1^{k+1} \odot \mathbf{A}_1^{k+1})\mathbf{1}_L \\
&\qquad\qquad \cdots (\mathbf{B}_R^{k+1} \odot \mathbf{A}_R^{k+1})\mathbf{1}_L]^T\|_F^2,
\end{aligned}$$

where $\mathbf{X}_{(1)} \in \mathbb{R}^{I\times JK}$, $\mathbf{X}_{(2)} \in \mathbb{R}^{J\times IK}$ and $\mathbf{X}_{(3)} \in \mathbb{R}^{K\times IJ}$ are the mode-1, mode-2 and mode-3 matricizations of tensor $\mathcal{X}$.

Given the initials $\mathbf{A}^0$, $\mathbf{B}^0$ and $\mathbf{C}^0$, then at the $(k+1)th$ iteration, we hold $\mathbf{B}^k$ and $\mathbf{C}^k$ to update the factor $\mathbf{A}$ to get $\mathbf{A}^{k+1}$, then $\mathbf{A}^{k+1}$ and $\mathbf{C}^k$ are held to

update $\mathbf{B}$ and obtain $\mathbf{B}^{k+1}$. Similarly, we hold $\mathbf{A}^{k+1}$ and $\mathbf{B}^{k+1}$ to obtain $\mathbf{C}^{k+1}$. Usually, the Frobenius norm of the error between the given tensor and the updated tensor is measured at each iteration to provide a stopping criterion.

There are some disadvantages to alternating least-squares algorithm (see [16], [19]). This method is not guaranteed to converge to a global minimum or even a stationary point of the cost function (4.6), only to a solution where the objective function ceases to decrease. Another issue of this method is that sometimes it needs a high number of iterations to converge, creating a swamp. In order to remove the swamp, [20] introduced a regularization method with the regularization parameter λ_n. The new algorithm for BTD-$(L, L, 1)$ is called BTD-RALS method.

We add the regularization item for each subproblem in the above three equations,

$$\begin{aligned}
\mathbf{A}^{n+1} &= \underset{\widehat{\mathbf{A}}\in\mathbb{R}^{I\times LR}}{\operatorname{argmin}} \|\mathbf{X}_{(\mathbf{1})} - \widehat{\mathbf{A}}(\mathbf{C}^n \odot_p \mathbf{B}^n)^T\|_F^2 \\
&\quad +\lambda_n\|\widehat{\mathbf{A}} - \mathbf{A}^n\|_F^2, \\
\mathbf{B}^{k+1} &= \underset{\widehat{\mathbf{B}}\in\mathbb{R}^{J\times LR}}{\operatorname{argmin}} \|\mathbf{X}_{(\mathbf{2})} - \widehat{\mathbf{B}}(\mathbf{C}^n \odot_p \mathbf{A}^{n+1})^T\|_F^2 \\
&\quad +\lambda_n\|\widehat{\mathbf{B}} - \mathbf{B}^n\|_F^2, \\
\mathbf{C}^{k+1} &= \underset{\widehat{\mathbf{C}}\in\mathbb{R}^{K\times R}}{\operatorname{argmin}} \|\mathbf{X}_{(\mathbf{3})} - \widehat{\mathbf{C}}[(\mathbf{B_1}^{n+1} \odot \mathbf{A_1}^{n+1})\mathbf{1}_L, \ldots, \\
&\quad (\mathbf{B_R}^{n+1} \odot \mathbf{A_R}^{n+1})\mathbf{1}_L]^T\|_F^2 + \lambda_n\|\widehat{\mathbf{C}} - \mathbf{C}^n\|_F^2.
\end{aligned}$$

In alternating fashion, these three subproblems are solved for the factor matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. The regularization parameters λ_n, $n = 0, 1, \ldots$, are given by a nonnegative decreasing sequence and at each iteration the parameters are the same for the three updated factor matrices. The rules for choosing the regularization parameters is also discussed in [20]. The regularized algorithm is summarized in the following Table 1. The number of iterations N in the algorithm is set to a large number, and a stopping criterion can be used.

Numerically, such regularized method is more efficient than the BTD-ALS in terms of reducing the number of iterations and accelerating the convergence.

Example. [Numerical example with a swamp] Here we give an example to show the swamp reducing technique of the BTD-RALS.

We create a tensor $\mathcal{X} \in \mathbb{R}^{10\times 15\times 28}$, which satisfies a block term decomposition in rank-$(3, 3, 1)$ with $R = 3$. The factor matrices are $\mathbf{A} \in \mathbb{R}^{10\times 9}$, $\mathbf{B} \in \mathbb{R}^{15\times 9}$ and $\mathbf{C} \in \mathbb{R}^{28\times 3}$, and the factorization equation is

$$\mathbf{X} = \sum_{r=1}^{3}(\mathbf{A}_r \cdot \mathbf{B}_r^{\mathrm{T}}) \circ \mathbf{c}_r.$$

RBTD-$(L, L, 1)$-Algorithm [20]

procedure RBTD-$(L, L, 1)(\mathcal{X}, R, N, \lambda_n)$

give initial guess $\mathbf{A}^0 \in \mathbb{R}^{I\times R}$, $\mathbf{B}^0 \in \mathbb{R}^{J\times R}$, $\mathbf{C}^0 \in \mathbb{R}^{K\times R}$, λ_0

for $n = 1, \ldots, N$ **do**

$\mathbf{W} \leftarrow [(\mathbf{C}^n \odot_p \mathbf{B}^n); \lambda_n\mathbf{I}^{LR\times LR}] \in \mathbb{R}^{(JK+LR)\times LR}$

$\mathbf{S} \leftarrow [\mathbf{X}_{(1)}; \lambda_n(\mathbf{A}^n)^T] \in \mathbb{R}^{(JK+LR)\times I}$

$\mathbf{A}^{n+1} \leftarrow \mathbf{W}/\mathbf{S}$ —— % solving least squares to update $\mathbf{A}$

$\mathbf{W} \leftarrow [(\mathbf{C}^n \odot_p \mathbf{A}^{n+1}); \lambda_n\mathbf{I}^{LR\times LR}] \in \mathbb{R}^{(IK+LR)\times LR}$

$\mathbf{S} \leftarrow [\mathbf{X}_{(2)}; \lambda_n(\mathbf{B}^n)^T] \in \mathbb{R}^{(IK+LR)\times J}$

$\mathbf{A}^{n+1} \leftarrow \mathbf{W}/\mathbf{S}$ —— % solving least squares to update $\mathbf{B}$

$\mathbf{W} \leftarrow [((\mathbf{B}_1^{n+1} \odot \mathbf{A}_1^{n+1})\mathbf{1}_L, \ldots, (\mathbf{B}_R^{n+1} \odot \mathbf{A}_R^{n+1})\mathbf{1}_L); \lambda_n\mathbf{I}^{R\times R}] \in \mathbb{R}^{(IJ+R)\times R}$

$\mathbf{S} \leftarrow [\mathbf{X}_{(3)}; \lambda_n(\mathbf{C}^n)^T] \in \mathbb{R}^{(IJ+R)\times K}$

$\mathbf{C}^{n+1} \leftarrow \mathbf{W}/\mathbf{S}$ —— % solving least squares to update $\mathbf{C}$

end for

return $\mathbf{A}^N$, $\mathbf{B}^N$, $\mathbf{C}^N$

end procedure

Table 1: **Regularized algorithm of BTD-$(L, L, 1)$ with rank R for a third-order tensor $\mathcal{X} \in \mathbb{R}^{I\times J\times K}$**

We use the same random initials for both BTD-ALS and BTD-RALS methods. Figure 3 shows that the BTD-RALS algorithm only takes 1558 iterations to reach an error within 10^{-4}, however, the BTD-ALS algorithm does not decrease the error after 20,000 iterations.

4.1 Convergence Property of RALS The regularized method of ALS (RALS) [20] for solving tensor CP decomposition and the convergence property of RALS have been studied in [19]. We have pointed out that the decomposition BTD-$(L, L, 1)$ is a general case of CP decomposition. In this section, we will show that the BTD-RALS has the same framework with the RALS and thus, the BTD-RALS has the same convergence property.

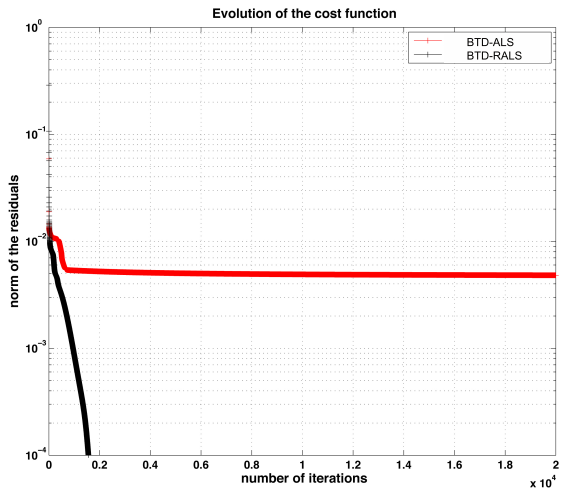


Figure 3: **The comparison of the BTD and RBTD with the same initials**

We can view the BTD-$(L, L, 1)$ problem as a nonlinear optimization. Problem (4.6) has the following expression,

$$\begin{aligned}(4.7)\quad &\min_{\mathbf{A},\mathbf{B},\mathbf{C}} f(\mathbf{A},\mathbf{B},\mathbf{C}) \\ &= \min_{\mathbf{A},\mathbf{B},\mathbf{C}} \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left[x_{ijk} - \sum_{r=1}^{R}\left(\sum_{l=1}^{L} a_{il}^{(r)} b_{jl}^{(r)}\right) c_k^{(r)}\right]^2,\end{aligned}$$

where $\mathbf{A} = [\mathbf{A}_1 \cdots \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1 \cdots \mathbf{B}_R]$ and $\mathbf{C} = [\mathbf{c}_1 \cdots \mathbf{c}_R]$ are the factor matrices. $a_{il}^{(r)}$ denotes the il element (ith row and lth column) of the matrix $\mathbf{A}_r$, $b_{jl}^{(r)}$ expresses the jl element of the matrix $\mathbf{B}_r$, and $c_k^{(r)}$ is the kth element in the vector $\mathbf{c}_r$.

So the cost function can be seen as a function from $\mathbf{y}$ to $\mathbb{R}$, where

$$\mathbf{y} = vec([vec(\mathbf{A})\ vec(\mathbf{B})\ vec(\mathbf{C})]) \in \mathbb{R}^n,$$

with $n = (IL + JL + K)R$.

Let $vec(\mathbf{A}) = \mathbf{y}_1 \in \mathbb{R}^{ILR}$, $vec(\mathbf{B}) = \mathbf{y}_2 \in \mathbb{R}^{JLR}$ and $vec(\mathbf{C}) = \mathbf{y}_3 \in \mathbb{R}^{KR}$. We partition the vector $\mathbf{y} \in \mathbb{R}^n$ into 3 component vectors $\mathbf{y}_i \in \mathbb{R}^{n_i}$, $i = 1, 2, 3$, where $n_1 = ILR$, $n_2 = JLR$ and $n_3 = KR$. It follows that $\mathbf{y} = \mathbf{y}_1 \times \mathbf{y}_2 \times \mathbf{y}_3 \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \mathbb{R}^{n_3} = \mathbb{R}^n$. Thus, the BTD-$(L, L, 1)$ can be reformulated to the following problem,

$$\begin{aligned}&\textbf{minimize} && f(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3), \\ &\textbf{subject to} && \mathbf{y} \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \mathbb{R}^{n_3} = \mathbb{R}^n.\end{aligned}$$

Therefore, the BTD-RALS method for solving the BTD-$(L, L, 1)$ updates each component of $\mathbf{y}$ in turn. Starting from a given initial point $\mathbf{y}^0 = vec([vec(\mathbf{A}^0)\ vec(\mathbf{B}^0)\ vec(\mathbf{C}^0)]) \in \mathbb{R}^n$, a sequence $\{(\mathbf{y}_1^k, \mathbf{y}_2^k, \mathbf{y}_3^k)\}$ is generated by the following equations

$$\begin{aligned}\mathbf{y}_1^{k+1} &= \underset{\mathbf{z}\in\mathbb{R}^{n_1}}{\operatorname{argmin}}\{f(\mathbf{z}, \mathbf{y}_2^k, \mathbf{y}_3^k) + \lambda_k\|\mathbf{y}_1^k - \mathbf{z}\|^2\}, \\ \mathbf{y}_2^{k+1} &= \underset{\mathbf{z}\in\mathbb{R}^{n_2}}{\operatorname{argmin}}\{f(\mathbf{y}_1^{k+1}, \mathbf{z}, \mathbf{y}_3^k) + \lambda_k\|\mathbf{y}_2^k - \mathbf{z}\|^2, \\ \mathbf{y}_3^{k+1} &= \underset{\mathbf{z}\in\mathbb{R}^{n_3}}{\operatorname{argmin}}\{f(\mathbf{y}_1^{k+1}, \mathbf{y}_2^{k+1}, \mathbf{z}) + \lambda_k\|\mathbf{y}_3^k - \mathbf{z}\|^2\}.\end{aligned}$$

Thus, the BTD-RALS algorithm has the same framework as the RALS for CP decomposition [19]. Moreover, the convergence property of RALS is applied to the BTD-RALS.

For the BTD-RALS method, we have the following theorem,

THEOREM 4.1. *Suppose that the sequence* $\{(\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)\}$ *obtained from the BTD-RALS has limit points. Then every points* $(\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})$ *is a critical point of the problem (4.7).*

Recall that a sequence that has a limit point $(\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})$ means that there exists a subsequence of $\{(\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)\}$ that converges to $\{(\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)\}$. The following is a critical point definition found in [1]. The critical point $(\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})$ of the problem (4.7) is a point such that

$$\bigtriangledown f(\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})^T \Big((\mathbf{A}, \mathbf{B}, \mathbf{C}) - (\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})\Big) \geq 0,\ \forall (\mathbf{A}, \mathbf{B}, \mathbf{C}).$$

According to this theorem, we can see that for a non-degenerate BTD problem [16], [19], the limit points obtained from the BTD-RALS method are the critical point of the original cost function (3.3).

5 Experiments

In this section, the air pollution collected data [24] is analyzed in rank-$(9, 9, 1)$ BTD using the BTD-RALS algorithm. Several figures are then created from the numerical results which explain the sources' identification via the sources profiles and time series of the source contributions. We also provide some numerical simulations on randomly generated tensor noisy data.

5.1 Environmental Data The original data was collected from 25 February to 10 April 2002 at Allen Park in Detroit, Michigan. The data was then sampled by using a three-stage Davis Rotating-drum Universal-size-cut Monitoring (DRUM) impactor [23], [24]. The particles were collected in three size modes, $2.5\mu m > D_p > 1.15\mu m$, $1.15\mu m > D_P > 0.34\mu m$ and $0.34\mu m > D_P > 0.1\mu m$, where D_p denotes the particle diameter. The air sample was also analyzed for elements of higher

atomic number. The 27 chemical species found were Na, Mg, Al, Si, P, S, Cl, K, Ca, Ti, V, Cr, Mn, Fe, Co, Ni, Cu, Zn, Ga, As, Se, Br, Rb, Sr, Zr, Mo, and Pb.

The data we study is considered as a function of size, time and chemical composition (a.k.a. elemental species). If we use i to denote chemical species, j to express particle size and k to be the time sample, then a data point x_{ijk} can be expressed as the concentration value of the ith chemical species of the jth particle size of the kth time sample.

See Figure 4 for the air sample tensor picture.

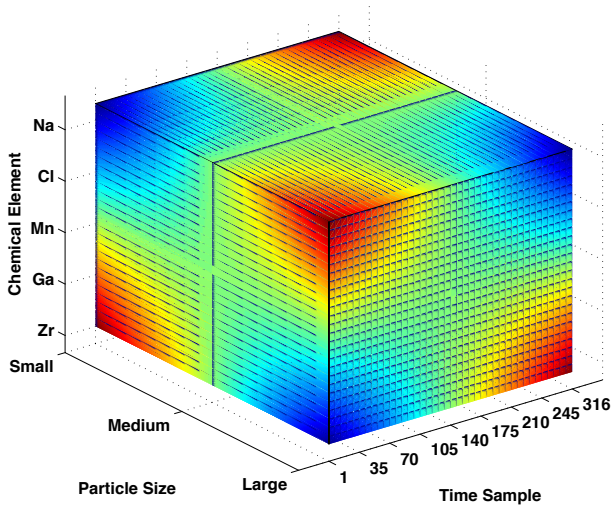


Figure 4: **The sampled data are placed on the three planes to construct the air tensor $\mathcal{X}$. Particle size denotes mode-J, Chemical composition denotes mode-I and Time sample denotes mode-K. So, the element x_{ijk} in $\mathcal{X}$ is the concentration value of the ith chemical species of the jth particle size range of the kth time sample.**

There are 27 chemical species and three different size of particle (small, medium, large), and according to [24], the time sample are 316 (3 time samples at first day; 8 time points on each day from the second to the last second day; 1 time point in the last day). So, the air tensor is a third-order tensor of size $27 \times 3 \times 316$.

5.2 Model Description According to [24], in order to separate the different factor sources from the dataset, the main equation is as follows:

$$x_{ijk} = \sum_{p=1}^{P} a_{ijp} b_{kp} + e_{ijk}, \tag{5.8}$$

where the x_{ijk} is the element of the third-order tensor obtained from the previous section, i.e., the concentration value of the ith chemical species of the jth particle size range of the kth time sample. In the elemental form, b_{kp} is the pth source mass contribution during the time units for the kth sample, a_{ijp} is the ith species mass fraction of the jth particle size range from the pth source. e_{ijk} is the residual associated with the ith species concentration measured in the kth sample of the jth size range, and P is the total number of independent sources.

Since each entry x_{ijk} denotes the concentration, a_{ijp} is the species mass fraction and b_{kp} is the source contribution, then we need add non-negativity constraints on a_{ijp} and b_{kp} when the tensor decomposition is applied.

We see that in the equation (5.8), x_{ijk} is an element of the air sample tensor $\mathcal{X}$. For each fixed p, a_{ijp} is an entry of a matrix $\mathbf{A}_p$ and b_{kp} is an element of a vector $\mathbf{b}_p$. Finally, the error tensor can be denoted as $\mathcal{E}$. Therefore, the equation (5.8) is equivalent to the following form,

$$\mathcal{X} = \sum_{p=1}^{P} \mathbf{A}_p \circ \mathbf{b}_p + \mathcal{E}. \tag{5.9}$$

Since $\mathcal{X} \in \mathbb{R}^{27\times 3\times 316}$, for each matrix $\mathbf{A}_p \in \mathbb{R}^{27\times 3}$, it can be decomposed into two matrices $\mathbf{C}_p \in \mathbb{R}^{27\times L_p}$ and $\mathbf{B} \in \mathbb{R}^{3\times L_p}$ so that $\mathbf{A}_p = \mathbf{C}_p \cdot \mathbf{D}_p^T$, where L_p is the rank of matrix $\mathbf{A}_p$. Therefore, the model also can be written as follows,

$$\mathcal{X} = \sum_{p=1}^{P} (\mathbf{C}_p \cdot \mathbf{D}_p^T) \circ \mathbf{b}_p + \mathcal{E}. \tag{5.10}$$

Thus, our goal is to find the matrices $\mathbf{C} = [\mathbf{C}_1 \ \cdots \ \mathbf{C}_P]$, $\mathbf{D} = [\mathbf{D}_1 \ \cdots \ \mathbf{D}_P]$ and $\mathbf{B} = [\mathbf{b}_1 \ \cdots \ \mathbf{b}_P]$ to minimize the following function,

$$Q(\mathbf{C}, \mathbf{D}, \mathbf{B}) = \left\| \mathcal{X} - \sum_{p=1}^{P} (\mathbf{C}_p \cdot \mathbf{D}_p^T) \circ \mathbf{b}_p \right\|_F^2. \tag{5.11}$$

This is exactly the error function for BTD (3.3) in rank-$(L_r, L_r, 1)$. When $L_i = L_j$, $1 \le i, j \le R$, it is the error function for BTD in rank-$(L, L, 1)$.

Since the BTD-$(L_r, L_r, 1)$ matches the model for the air dataset, we can use the algorithm BTD-RALS to solve for the air sample tensor to minimize the above function (5.11).

In [24], the cost function used is

$$Q = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\frac{x_{ijk} - \sum_{p=1}^{P} a_{ijp} b_{kp}}{u_{ijk}} \right)^2, \tag{5.12}$$

where u_{ijk} is an uncertainty estimate element in the ith species of the jth particle size of the kth time sample. The procedure of Polissar [22] was used to assign measured data and the associated uncertainties as the input data.

Comparing the functions (5.12) (element-wise) and (5.11), the cost function in our method does not include the uncertainty estimates. We will use the cost function without the uncertainties u_{ijk} and consider the function (5.12) as a constraint on our solution.

According to the paper [24], there are 9 sources. Thus, in the BTD model (5.11), we let $P = 9$. For the tensor $\mathcal{X} \in \mathbb{R}^{27\times 3\times 316}$, the block term decomposition in rank-$(L_r, L_r, 1)$ with 9 terms is not essentially unique (see [6]). Recall that essentially uniqueness indicates the decomposition is unique up to permutation and scaling. Since the tensor data does not satisfy the uniqueness criteria, then we will have multiple solutions from the algorithm. We tested a large number of initial conditions with different L_p, $1 \le p \le 9$ and found that under the setting of $L_p = 9$, $p = 1, 2, \ldots, 9$, we can find a solution that is consistent with the numerical results found in [24].

For the non-negativity constraints on the decomposition, we need to use the BTD-RALS method to solve for the non-negative factor matrices in the problem (5.12). So we will add non-negativity constraints on the three subproblems. In terms of solving these subproblems, for each least-squares problem with non-negativity constraints, we can use the method in [17], or we can use the non-negative matrix factorization (NMF) introduced by Lee and Seung [18] to solve the each subproblem with constraint.

In this paper, we use the algorithm in Table 1 with the method by [17] to obtain the three non-negative factor matrices $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{B}$ in the cost function (5.11).

5.3 Result and Discussion We apply the BTD-$(9, 9, 1)$ on the air sample tensor $\mathcal{X}$ (5.11). By using the BTD-RALS algorithm with non-negativity constraints, we can obtain the three factor matrices

$$\mathbf{C} = [\mathbf{C}_1 \ \cdots \ \mathbf{C}_9], \ \ \mathbf{D} = [\mathbf{D}_1 \ \cdots \ \mathbf{D}_9], \ \ \mathbf{B} = [\mathbf{b}_1 \ \cdots \ \mathbf{b}_9],$$

where $\mathbf{C}_p \in \mathbb{R}^{27\times 9}$, $\mathbf{D}_p \in \mathbb{R}^{3\times 9}$ and $\mathbf{B} \in \mathbb{R}^{316\times 9}$, $p = 1, 2, \ldots, 9$. Therefore, according to the model for the air sample, for each p, $\mathbf{A}_p = \mathbf{C}_p \cdot \mathbf{D}_p^T \in \mathbb{R}^{27\times 3}$ is a matrix and the elements of such matrix are exactly the a_{ijp}s (for a fixed p) in the equation (5.8). Furthermore, each $\mathbf{A}_p$ denotes a source profile. So, for each $\mathbf{A}_p$ we have a bar plot for one source (see the Figure 5a). The vector $\mathbf{b}_p$ in the factor matrix $\mathbf{B}$ is the vector b_{kp} (for a fixed p) in the model equation (5.8). It denote the contribution of p source in terms of time. Therefore, the Figure 5b expresses the time series of the source contributions.

From the source profile bar plot 5a, we can figure out the nine different factor sources, they are: Industrial (Fe+Zn), Sulfur with Dust, Road Dust, two types of Metal Works, Local Sulfate, Road Salt, Homogeneously formed Sulfate and Cloud Processed Sulfate. For the explanation for each source, refer to the paper [24]. In the Figure 5b, we can also see the change of each source contribution in time. For example, there are several spikes in the time series of the road source contribution. This indicates additional snowfall or low temperatures where the ice melting. Furthermore, we can also tell the contribution change of other factor sources.

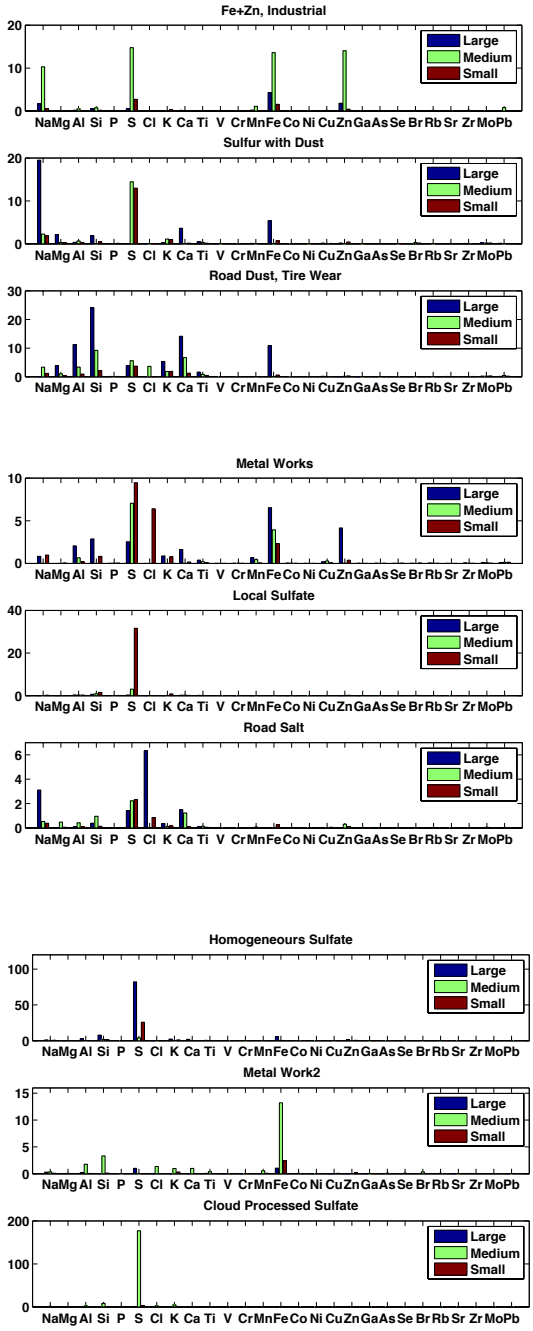
The identification from the bar plots in Figure 5a is also based the particle sizes. This method provides a more accurate result comparing the classical matrix factorization method. It is seen that in the industrial source, the high concentrations of Fe and Zn are in the middle size rage while the large size rage of Fe and Zn are shown in the metal works.

We can also analyze the concentration during the weekdays and weekends for each factor source (see the Figure 6). The left nine plots show the concentration change during the weekdays and the corresponding right plots show the concentration change during the weekends for each factor source. In each day, we take the maximum and minimum concentration at the time points 1, 4, 7, 10, 13, 16, 19, 22 and then take the average for each time point. Figure 6 shows the factors concentrations of Road Salt, Homogeneous Sulfate and the second type of Metal Work are high during the weekdays and lower during the weekends. This is very reasonable since the weekdays are typically work days when industrial companies are in operation. There is also a longer commuting hours during the weekdays than the weekends, explaining the higher concentration of Road Salt. For some factors like local sulfate, there is no difference between the weekdays and weekends which indicates the people's activities are not a major effect on these environmental factors.

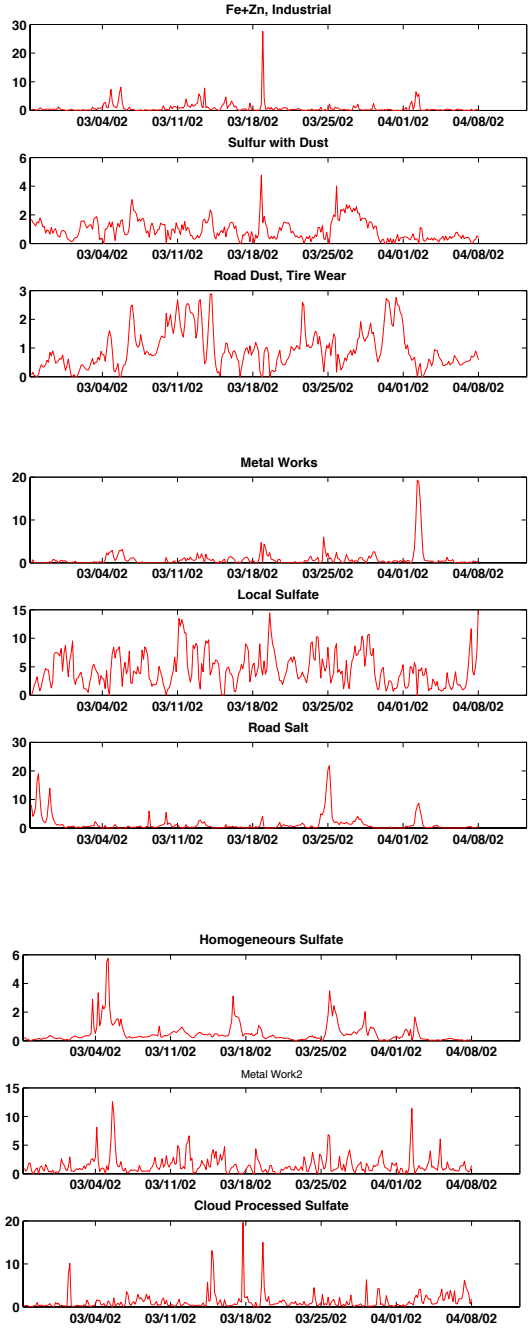
With the Figures 5a, 5b and 6 and above discussion, we show that the BTD-RALS method can successfully identify the nine different factor sources and also provides the time contribution of each factor.

5.4 BTD-RALS on the noisy data Here we give a numerical experiment to show the BTD-RALS method works on the third-order tensor data with noise. We generate tensors $\widetilde{\mathcal{T}} \in \mathbb{R}^{5\times 6\times 7}$ in the following way:

$$\widetilde{\mathcal{T}} = \frac{\mathcal{T}}{\|\mathcal{T}\|} + \sigma_N \frac{\mathcal{N}}{\|\mathcal{N}\|}, \tag{5.13}$$



(a) Source profiles for the resolved factors. The y-axis denotes the relative elemental concentration. The order of the particle size is Large, Medium, Small from left to right for each element.



(b) The time series of source contributions.

Figure 5: **Nine factor sources identified from the air sample.**

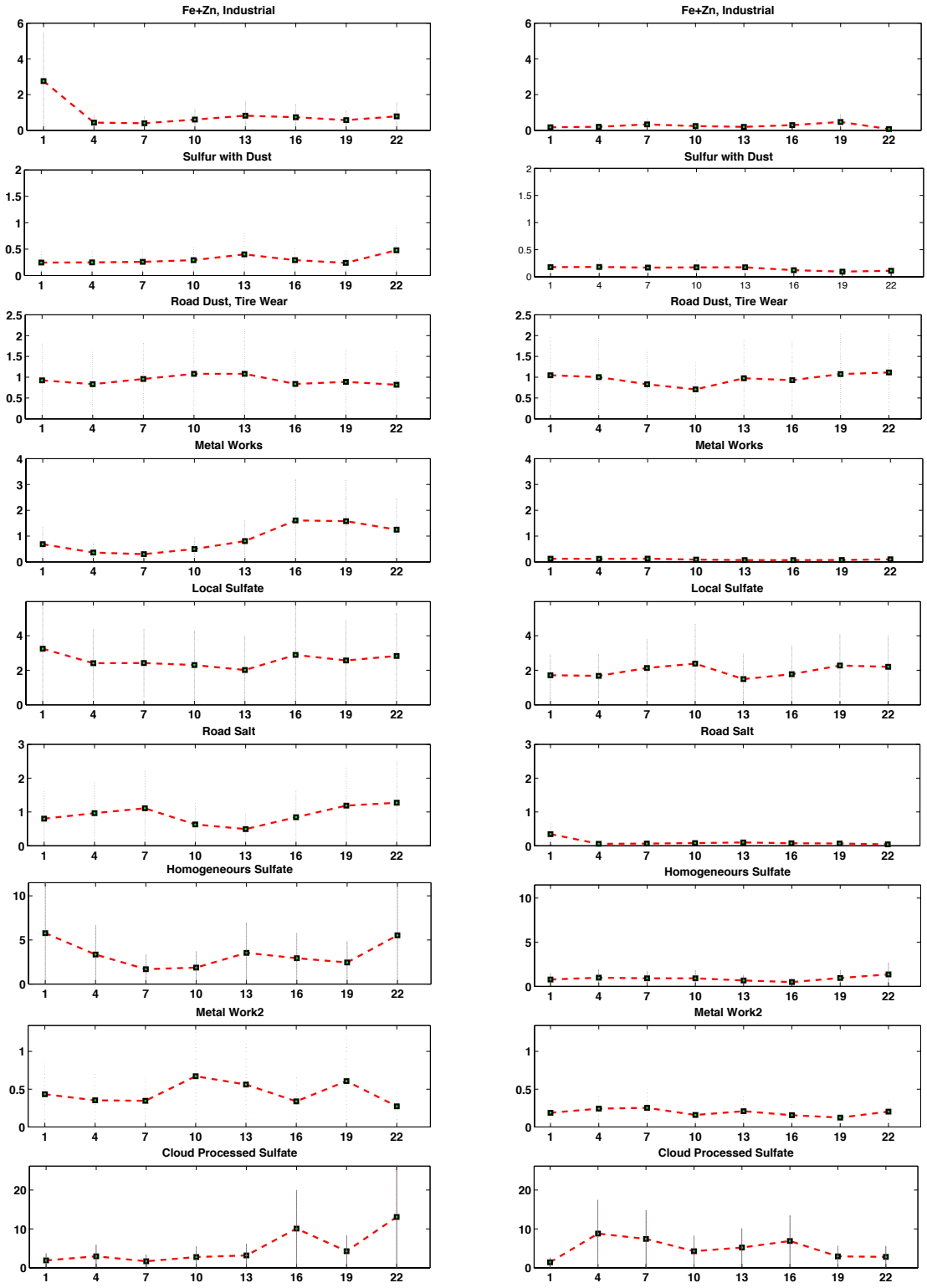


Figure 6: **Time of a day variations. The left column shows the concentration change in weekdays for each factor source. The right column shows the concentration change in weekends for each factor source**

where $\mathcal{T}$ has the block term decomposition in rank-$(2,2,1)$ with $R = 3$ in the equation (3.2) so that $\mathbf{A}_r \in \mathbb{R}^{5\times 2}$, $\mathbf{B}_r \in \mathbb{R}^{6\times 2}$ and $\mathbf{c}_r \in \mathbb{R}^{7\times 1}$, $r = 1,2,3$. The second term in (5.13) is the noise term and the parameter σ_N controls the noise level. The entries of $\mathbf{A} = [\mathbf{A}_1\ \mathbf{A}_2\ \mathbf{A}_3]$, $\mathbf{B} = [\mathbf{B}_1\ \mathbf{B}_2\ \mathbf{B}_3]$, $\mathbf{C} = [\mathbf{c}_1\ \mathbf{c}_2\ \mathbf{c}_3]$ and the tensor $\mathcal{N}$ are drawn from a zero-mean unit-variance Gaussian distribution.

By using BTD-RALS method, a Monte Carlo experiment of BTD-$(2,2,1)$ with $R = 3$ on $\widetilde{\mathcal{T}}$ consisting 50 runs is carried out. The algorithm is initialized with three random starting values.

We measure the relative error $e = \|\mathbf{C} - \widetilde{\mathbf{C}}\|/\|\mathbf{C}\|$, where $\widetilde{\mathbf{C}}$ is the approximation of $\mathbf{C}$, optimally permuted and scaled. The median results are plotted in Figure 7. It is seen that with low noise levels, average error in $\mathbf{C}$

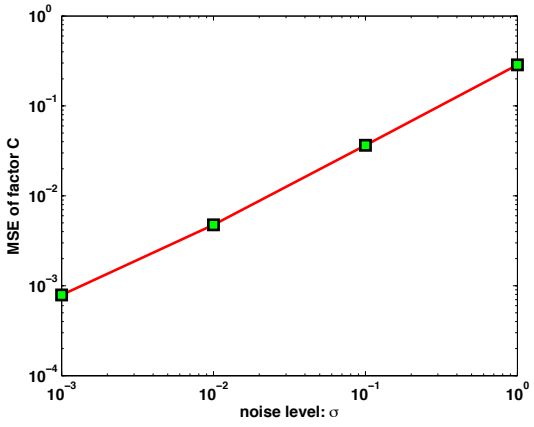


Figure 7: Median relative error v.s. noise level

increases proportionally to noise level.

6 Conclusion

The method BTD-RALS is presented in the application of identifying the factor sources of the collected air sample. The dataset is formed into a third order tensor and the data model is written into a block term decomposition in rank-$(L,L,1)$. We apply a regularized alternating method to solve the block term decomposition and obtain the resulting factor matrices providing the source profiles and source contributions correctly. In addition, we show that regularized method is efficient than the classical alternating method numerically and can converge to a stationary point of the original BTD cost function under some assumption. The BTD-RALS algorithm is also tested on the random data with different noise levels, which shows the average error in the third factor matrices $\mathbf{C}$ increases proportionally to noise level.

References

[1] D. P. Bertsekas, *Nonlinear Programming,* Athena Scientific, Belmont, MA, 1995.

[2] R. Bro, *Multi-way analysis in the food industry: models, algorithms, and applications*, Ph.D. thesis, University of Amsterdam, 1998.

[3] R. Bro, *PARAFAC. Tutorial and applications*, Chemometrics and Intelligent Laboratory Systems, 38 (1997), pp. 149-171.

[4] J. Carrol and J. Chang, *Analysis of Individual Differences in Multidimensional Scaling via an N-way Generalization of "Eckart-Young" Decomposition*, Psychometrika, 9 (1970), pp. 267-283.

[5] L. De. Lathauwer, *Decompositions of a higher-order tensor in block terms – part I: lemmas for partitioned matrices*, SIAM J. Matrix Anal. Appl., 30 (2008), No. 3, pp. 1022-1032.

[6] L. De. Lathauwer, *Decompositions of a higher-order tensor in block terms – part II: definitions and uniqueness*, SIAM J. Matrix Anal. Appl., 30 (2008), No. 3, pp. 1033-1066.

[7] L. De. Lathauwer and N. Dimitri, *Decompositions of a higher-order tensor in block terms – part III: alternating least squares algorithms.* SIAM J. Matrix Anal. Appl., 30 (2008), No. 3, pp. 1067-1083.

[8] L. De Lathauwer and J. Vandewalle, *Dimensionality reduction in higher-order signal processing and rank-$(R_1, R_2, \cdots, R_N)$ reduction in multilinear algebra*, Linear Algebra Appl., 391 (2004), pp. 31-55.

[9] L. Grippo and M. Sciandrone, *On the convergence of the block nonlinear Gauss-Seidel method under convex constraints*, Operations Research Letters, 26 (2000), pp. 127-136.

[10] R. A. Harshman, *Foundations of the PARAFAC procedure: Model and Conditions for an "Explanatory" Multi-code Factor Analysis*, UCLA Working Papers in Phonetics, 16 (1970), pp. 1-84.

[11] R. Henrion, *body diagonalization of core matrices in three-way principal components analysis: Theoretical bounds and simulation*, J. Chemometrics, 7 (1993), pp. 477-494.

[12] R. Henrion, *N-way principal component analysis theory, algorithms and applications*, Chemometrics and Intelligent Lboratory Systems, 25 (1994), pp. 1-23.

[13] I.J. Hwang, P.K. Hopke and J.P. Pinto, *Source Apportionment and Spatial Distributions of Coarse Particles During the Regional Air Pollution Study (RAPS)*, Environ. Sci. Technol., 42 (2008), pp. 35243530.

[14] P.K. Hopke, *Receptor Modeling in Environmental Chemistry*, Wiley, New York, 1985.

[15] E. Kim and P.K. Hopke, *Source Characterization of Ambient Fine Particles at Multiple Sites in the Seattle Area, , Atmospheric Environ.,* 42 (2008), pp. 6047-6056.

[16] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Review., 51 (2009), No.3, pp. 455-500.

[17] C.L. Lawson and R.J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, 23 (1974), pp. 161.

[18] D. Lee and H. S. Seung, *Algorithms for non-negative matrix factorization*, in Advances in Neural Information Processing 3 (Proc. NIPS 2000), MIT Press, 2001.

[19] N. Li, S. Kindermann and C. Navasca, *Some convergence results on the regularized alternating least-squares method for tensor decomposition*, submitted.

[20] C. Navasca, L. De Lathauwer and S. Kindermann, *Swamp reducing technique for tensor decomposition*, in the 16th Proceeding of the European Signal Processing Conference, Lausanne, August 2008.

[21] P. Paatero, *The multilinear engine–a table–driven least squares program for solving multilinear problems, including the n–way parallel factor analysis model*, Journal of Computational and Graphical Statistics, 8 (1999), pp. 854-888.

[22] A.V. Polissar, P.K. Hopke, P. Paatero, W.C. Malm and J.F. Sisler, *Atmospheric aerosol over alaska 2. Elemental composition and sources.* Journal of Geophysical Research, 103 (1998), 19045-19057.

[23] O.G. Raabe, D.A. Braaten, R.L. Axelbaum, S.V. Teague, T.A. Cahill, *Calibration studies of the DRUM impactor*, Journal of Aerosol Science, 19 (10988), pp. 183-195.

[24] E. Peré-Trepat, E. Kim, P. Paatero and P.K. Hopke, *Source apportionment of time and size resolved ambient particulate matter measured with a rotating DRUM impactor*, Atmospheric Environment, 41 (2007), pp. 5921-5933.

[25] L. R. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966), pp. 279-311.

[26] L. R. Tucker, *Implications of factor analysis of three-way matrices for measurement of change*, in Problems in Measuring Change, C. W. Harris, ed., University of Wisconsin Press, 1963.